

# Estudo comparativo de classificadores aplicado à análise de cenas de pomares de laranjeiras

Felipe A. Cavani  
USP - EESC - SEM  
Av. Trabalhador São-carlense, 400  
São Carlos, São Paulo, Brasil  
fcavani@gmail.com

Mário Luiz Tronco  
UNESP - IBILCE - DCCE  
R. Cristóvão Colombo, 2265  
S. J. do R. Preto, São Paulo, Brasil  
mariot@ibilce.unesp.br

Arthur José Vieira Porto  
USP - EESC - SEM  
ajvporto@sc.usp.br

## Resumo

*A análise de imagens agrícolas exige algoritmos com capacidade de lidar com cenas não estruturadas ao ar livre. Algoritmos de segmentação desenvolvidos para segmentar cenas naturais juntamente com classificadores de segmentos viabilizam sistemas de visão computacional onde o algoritmo de segmentação é distinto da classificação dos segmentos. Tal fato é explorado neste trabalho com o propósito de avaliar vetores de características e diferentes tipos de classificadores de padrões. Foram avaliados vetores de características derivados dos espaços de cores RGB e HSV. Também foram avaliados classificadores baseados no teorema de Bayes e baseados no perceptron multicamadas. Os classificadores bayesianos treinados com os vetores de características derivados do espaço de cores HSV mostraram-se mais adequados nos experimentos realizados.*

## 1. Introdução

Sistemas de visão computacional (VC), quando usados para analisar imagens provenientes de ambiente não estruturado ao ar livre, necessitam de algoritmos que devem ser robustos com relação as mudanças de iluminação, reflexos e sombras. Algoritmos de segmentação adequados para estes tipos de ambientes, juntamente com técnicas de classificação de padrões, possibilitam a construção de sistemas de VC adequados para estes ambientes.

Um sistema de VC onde um algoritmo de segmentação automático opera de modo independente da classi-

ficação supervisionada dos segmentos, pode ser considerado como de arquitetura modular. Tal arquitetura é vantajosa pois através desta é possível avaliar separadamente algoritmos de segmentação, descritores de segmentos e classificadores de padrões. O algoritmo de segmentação deve gerar segmentos com fronteiras bem definidas, coerentes com os objetos presentes na cena. Os descritores desempenham nesta arquitetura um papel fundamental, pois estes impactam diretamente na qualidade dos resultados obtidos do classificador. Por isto, devem ser escolhidos descritores adequados. Também, devem ser consideradas abordagens adequadas para a classificação destes descritores.

As vantagens desta arquitetura são adequadas para sistemas de VC agrícolas. Tais sistemas têm se mostrado importantes, pois a área agrícola tem usado tecnologias com o propósito de melhorar a competitividade ou aumentar a produção. Tais sistemas de VC têm sido usados para: inspeção de produtos, monitoramento da plantação e colheita automática [10, 15]. Quando considerada a cadeia produtiva de laranjas, a VC tem sido aplicada para coleta de dados [1, 14] ou para auxiliar a colheita automática de frutos [13], dentre outras aplicações.

Neste trabalho, o algoritmo de segmentação JSEG [3] é usado, pois este opera de modo não supervisionado, o que possibilita que o conhecimento a respeito do ambiente seja embutido na etapa de classificação dos segmentos. Posteriormente, os segmentos são classificados com o classificador de Bayes, classificador ingênuo de Bayes e classificador baseado no perceptron multicamadas (PMC) treinado com o algoritmo iRprop [8]. Os resultados obtidos com os classificadores, treinados com diferentes tipos de ve-

tores de características, são comparados. Por fim, são feitas considerações sobre o uso de tal arquitetura em aplicações reais.

Nas seções seguintes, este trabalho é detalhado e, por fim, os resultados e conclusões são apresentados.

## 2. Materiais

As imagens das laranjeiras foram adquiridas em diferentes dias e horários durante a época em que as frutas estavam maduras. A resolução destas imagens é de  $2.048 \times 1.536$  pixels. Um conjunto de 658 imagens forma o banco de imagens, que é dividido em três grupos: um que mostra as árvores desde o solo até o topo, outro que mostra apenas folhas e frutos e outro que mostra as fileiras de árvores, o solo e o céu. Estes três tipos de imagens são ilustradas pela figura 1.



**Figura 1. Tipos de imagens usadas neste trabalho.**

Os algoritmos usados neste trabalho foram implementados para o software de computação numérica Octave. Alguns dos algoritmos foram implementados em C++ como módulos de expansão do Octave. As funções de densidade de probabilidade (fdp) foram estimadas com o algoritmo de Figueiredo-Jain [5] implementado no *toolbox* desenvolvido por Paalanen *et al.* [12]. Também, os classificadores bayesianos são implementados com este *toolbox*. Os PMC foram construídas, treinadas e testadas com os módulos para o Octave da biblioteca Fann (*fast artificial neural network*).

## 3. Características dos segmentos

O uso das cores como característica é bastante comum em sistemas agrícolas de VC, pois estas podem caracterizar adequadamente os objetos de interesse, como por exemplo, as frutas [10, 15, 1, 14].

Neste trabalho, o descritor de cores adotado é o histograma. O cálculo do histograma das cores, onde são consideradas as três componentes, é muito custoso, pois a quantidade de cores é muito grande, quando as componentes são representadas por 256 níveis. Entretanto, é possível calcular o histograma de cada componente separadamente, o que é computacionalmente simples,

então, estes histogramas são unidos em um único vetor de características. Este vetor é capaz de representar o segmento de forma adequada, entretanto ele não retém a informação cromática.

Os histogramas, neste trabalho, são formados por 256 categorias para cada uma das componentes do espaço de cores, assim a informação contida em cada componente não é perdida. Esta abordagem difere da adotada por [9], que define números de categorias diferentes para cada uma das componentes do espaço de cores, o que acaba privilegiando alguma componente em detrimento das demais.

Os espaços de cores adotados foram o HSV e o RGB: o HSV faz distinção entre a informação cromática e a quantidade de luz, o que torna este espaço de cores mais robusto a situações de sombras e reflexos; já o RGB possibilita comparar o desempenho dos classificadores com ambos os tipos de vetores de características.

## 4. Conjuntos de treinamento e de teste

Foram escolhidos segmentos de 40 imagens. Estes segmentos foram classificados manualmente nas classes frutas, céu e folhas, que representam os principais elementos da cena. Os vetores de características foram calculados a partir da informação contida em cada *pixel* do segmento. Então, estes vetores foram separados em dois conjuntos. Um conjunto para o treinamento e outro para teste dos classificadores. O conjunto de teste é composto por 839 vetores selecionados aleatoriamente.

Diferentes conjuntos de treinamento foram criados a partir da partição do conjunto original. Somente, os vetores mais próximos à centróide da sua classe foram mantidos nos novos conjuntos. Foram considerados vetores próximos à centróide, os vetores que distam uma porcentagem da distância euclidiana entre o vetor mais distante desta classe e a centróide da mesma classe. As porcentagens usadas são 20%, 30%, 50%, 70% e 100%.

O número de vetores em cada classe para cada um dos conjuntos de treinamento é apresentado nas tabelas 1 e 2 para o caso RGB e HSV, respectivamente.

## 5. Extração e normalização das características

A PCA (*principal components analysis* – análise de componentes principais) é uma técnica que faz uma transformação linear no espaço de característica de modo a obter-se características não correlacionadas. Também, é possível com esta técnica reduzir o número de dimensões deste espaço.

Conjunto	Frutas	Folhas	Céu	Total
20%	1110	5602	35	6747
30%	1278	5997	117	7392
50%	1349	6283	225	7857
70%	1363	6297	261	7921
100%	1364	6297	275	7936

**Tabela 1. Número de vetores em cada classe para o caso RGB.**

Conjunto	Frutas	Folhas	Céu	Total
20%	566	5405	31	6002
30%	1155	6029	89	7273
50%	1331	6277	209	7817
70%	1363	6297	255	7915
100%	1364	6297	275	7936

**Tabela 2. Número de vetores em cada classe para o caso HSV.**

Com o objetivo de avaliar a relação entre quantidade de dimensões e a taxa de classificações corretas, esta técnica foi adotada. Deste modo também, evita-se a “maldição da dimensionalidade”, a qual prejudica diretamente os classificadores [6].

Neste trabalho, a PCA foi feita em cada um dos conjuntos de treinamento e as matrizes de transformação encontradas foram usadas para transformar os conjuntos de treinamento e de teste.

## 6. Classificador baseado no perceptron multicamadas

Classificadores construídos com redes neurais artificiais (RNA) toleram ruídos, são rápidos, devido ao seu baixo custo computacional, e podem aprender superfícies de decisão não-lineares [7]. O perceptron multicamadas (PMC) é uma das arquiteturas de RNA que pode ser usada como classificador de padrões, entretanto, o algoritmo padrão de treinamento, o backpropagation, é lento, ou seja, precisa de muitas iterações para convergir. O treino do PMC pode ser mais rápido com o algoritmo iRprop, um algoritmo adaptativo de descida de gradiente que é acurado, robusto e rápido [8].

Neste trabalho, cada tipo de vetor de características é classificado por um PMC com topologia específica, onde o número de entradas coincide com o número de dimensões dos vetores de características. Então, são construídos PMC para cada conjunto de treinamento, onde estes PMC suportam vetores de caracte-

rísticas de duas até 20 dimensões, o que resulta em 190 PMC. Todos os PMC têm 10 neurônios na camada escondida e três neurônios na camada de saída, tal que cada um destes representa uma das classes.

Todos os PMC são treinados três vezes. O treinamento termina quando o erro quadrático médio (EQM) é menor que 0,0001 ou o número de iterações é maior que 200.000. Antes de cada treinamento, os vetores são embaralhados. A função de ativação dos neurônios é a tangente hiperbólica, com parâmetro igual a 0,9. Os pesos sinápticos são inicializados com valores aleatórios. Os demais parâmetros têm os valores padrão da biblioteca Fann.

As redes, obtidas após o treinamento, são usadas para classificar os vetores do conjunto de teste. Cada vetor é apresentado para a rede e é atribuído para a classe correspondente ao neurônio de saída que retorná o maior valor. A taxa de acertos de cada classificador é calculada comparando-se os dados obtidos deste com os dados obtidos na classificação manual dos segmentos. Dentre os três treinamentos de cada uma das redes, o que apresenta maior taxa de acertos é escolhido.

## 7. Classificador baseado no teorema de Bayes

A abordagem estatística utiliza o teorema de Bayes para calcular o valor da probabilidade de um determinado vetor de características pertencer a uma classe. Isto possibilita a atribuição criteriosa a uma classe, quando a fdp e a probabilidade *a priori* modelam adequadamente o conjunto de treinamento. Caso a fdp e a probabilidade *a priori* não sejam conhecidas, estas devem ser estimadas a partir do conjunto de treinamento.

A fdp pode ser obtida através de técnicas paramétricas ou não-paramétricas. As técnicas paramétricas são utilizadas quando é conhecida a fdp que originou o conjunto de treinamento, mas os parâmetros desta não são conhecidos e devem ser estimados. As técnicas não-paramétricas são necessárias quando não existe conhecimento suficiente que possa ser utilizado para determinar a fdp ou quando a densidade do conjunto de treinamento assume uma função não conhecida, como é o caso deste trabalho.

As técnicas não-paramétricas podem ser baseadas em janelas ou em vizinhos mais próximos. Uma das dificuldades é encontrar um tamanho de janela adequado para todo o domínio do conjunto de treinamento. Os métodos de vizinhos mais próximos não sofrem deste problema, entretanto a quantidade de vizinhos também deve ser determinada manualmente. Os dados a serem avaliados na estimativa de um ponto da fdp po-

dem ser suavizados por meio de uma função de núcleo, como é o caso da função gaussiana, que atribui menor peso aos vetores de características que estão mais distantes do centro. Esta última técnica é chamada de não-paramétrica de núcleo [4].

Os modelos baseados em misturas finitas são semelhantes aos métodos não-paramétricos de núcleo e são capazes de representar fdp arbitrárias e complexas [6]. Neste caso, a função que modela o conjunto de treinamento é aproximada pela combinação de um número finito de funções, onde cada uma destas funções, ou componentes, é ponderada por um valor, o que dá origem a mistura. Estes modelos permitem uma abordagem formal ao aprendizado não supervisionado e, conseqüentemente, permitem que a escolha do número de componentes e a validação do modelo sejam feitas formalmente, diferente das heurísticas de agrupamento como o k-médias. As misturas são semelhantes aos modelos criados pelos algoritmos de agrupamento, pois cada componente da mistura é associado a um sub-conjunto (agrupamento) do conjunto de treinamento, onde este sub-conjunto têm maior probabilidade de ter sido gerado por este componente da mistura. A função gaussiana é normalmente escolhida para compor a mistura [5].

Neste trabalho, a fdp é aproximada como uma mistura de funções gaussianas e assume a forma da equação 1, onde  $k$  é o número de funções,  $\alpha_m$  são os pesos ou probabilidades das funções,  $\theta_m$  são os parâmetros da função gaussiana,  $\theta \equiv \{\theta_1, \dots, \theta_k, \alpha_1, \dots, \alpha_k\}$ ,  $\alpha_m \geq 0$  com  $m = 1, \dots, k$  e  $\sum_{m=1}^k \alpha_m = 1$  [5].

$$\hat{p}(y|\theta) = \sum_{m=1}^k \alpha_m p(y|\theta_m) \quad (1)$$

Determinar a fdp, ou o modelo do conjunto de treinamento, neste trabalho, está restrito a escolha do número de componentes da mistura ( $k$ ) e a encontrar os parâmetros destas componentes ( $\theta$ ). Estes parâmetros podem ser encontrados analiticamente, mas devido a falta de informação a respeito dos conjuntos de treinamento, como médias e matrizes de co-variâncias, esta abordagem não é adequada. Entretanto, o número de componentes e os parâmetros destas podem ser encontrados através de *maximum-likelihood* (ML) ou através da estimativa bayesiana. Na prática a ML é mais simples e produz resultados tão acurados quanto a estimativa bayesiana [11].

O algoritmo padrão para obter a ML é o algoritmo de maximização da esperança (EM - *expectation-maximization*). Entretanto, este algoritmo tem algumas deficiências, como, por exemplo, sensibilidade à inicialização e a necessidade da escolha do número de

componentes da mistura, onde este número leva a sub-adaptação ou a super-adaptação. Figueiredo e Jain [5] modificaram o algoritmo EM de forma que o novo algoritmo fosse capaz de lidar com estes problemas. Este novo algoritmo inicia com um número arbitrário e grande de componentes, onde as médias destas componentes são distribuídas pelo espaço formado pelo conjunto de treinamento. Estas componentes são eliminadas quando o algoritmo detecta que estas não são suportadas pelo conjunto de treinamento, criando, assim, um novo modelo. Após o algoritmo atingir um número mínimo de componentes, é escolhido, dentre todos os modelos avaliados, o modelo que tem a função de objetivo com menor valor [5].

O algoritmo de Figueiredo-Jain foi usado neste trabalho para estimar as fdp de cada classe a partir dos conjuntos de treinamento. Como é feito para os classificadores baseados no PMC, são estimadas fdp para cada uma das quantidades de dimensões, ou seja, de duas até 20 dimensões, de cada um dos conjuntos de treinamento, sendo que para cada um destes casos, as fdp são estimadas três vezes. Então, a classificação do conjunto de teste é feita com a equação 2, onde  $i$  é uma classe,  $P(C_i)$  é a probabilidade da classe  $C_i$  ocorrer,  $p(y|C_i)$  é fdp condicional da classe  $i$ ,  $y$  é o vetor de características e  $K$  é o número total de classes [2]. A classificação que resultar na maior taxa de acerto é escolhida e as duas outras são descartadas. Os parâmetros do algoritmo de Figueiredo-Jain são os padrões do *toolbox* [11].

$$P(C_i | y) = \frac{p(y | C_i)P(C_i)}{\sum_{j=1}^K p(y | C_j)P(C_j)} \quad (2)$$

## 8. Classificador ingênuo de Bayes

Este classificador é semelhante ao apresentado anteriormente, entretanto, neste caso, admite-se que as características são independentes. Então, para cada uma das classes, supondo que a fdp assume a forma da equação 3, onde  $p(y_j | C_i)$  é a função de densidade marginal condicional, e  $n$  é o número de características. Neste caso, a função de densidade marginal condicional foi também estimada com o algoritmo de Figueiredo-Jain.

$$p(y | C_i) = \prod_{j=1}^n p(y_j | C_i) \quad (3)$$

Classificadores que adotam a fdp desta forma podem lidar com mais características, pois cada fdp marginal terá apenas uma dimensão. Apesar da fdp deste classificador ser distorcida, ela não prejudica a classificação, pois a distorção é menor perto das fronteiras de decisão [6].

O treinamento e a avaliação deste classificador são feitos da mesma forma que no classificador baseado no teorema de Bayes.

## 9. Resultados

Os resultados obtidos após o treinamento dos vários classificadores são resumidos na tabela 3, onde são apresentados os melhores classificadores para cada tipo de classificador e para cada tipo de vetor de características (vetores derivados dos espaços de cores RGB ou HSV). Pode-se observar que os melhores classificadores são: o baseado no perceptron multicamadas e o baseado no teorema de Bayes. Entretanto, para o caso bayesiano o classificador obteve tal taxa utilizando menos dimensões e as fdp foram estimadas de um conjunto de treinamento menor, onde os vetores de tal conjunto estão menos dispersos no espaço. Considerando que os vetores mais distantes não foram removidos dos conjuntos de teste, é possível afirmar que o classificador bayesiano teve melhor capacidade de generalização.

Classificador	Cores	D	C	Taxa
MLP	RGB	18	100%	0,94
MLP	HSV	20	50%	0,95
Bayes	RGB	17	70%	0,94
Bayes	HSV	17	30%	0,95
Naive	RGB	20	100%	0,92
Naive	HSV	10	70%	0,92

**Tabela 3. Taxas de acerto dos melhores classificadores, onde D é o número de dimensões e C é o conjunto de treinamento.**

Para avaliar o desempenho dos classificadores com relação ao número de dimensões, são considerados os valores médios da taxa de acerto obtidos para os conjuntos de treinamento 50%, 70% e 100%. Os conjuntos 20% e 30% foram desconsiderados, pois a quantidade de vetores da classe céu é pequena. Devido a isto, não foi possível estimar as fdp com o algoritmo de Figueiredo-Jain para estes casos.

O gráfico na figura 2 mostra as taxas médias de acerto para todos os tipos de classificadores, espaços de cores e quantidade de dimensões. Neste gráfico é possível observar que os classificadores bayesianos treinados com vetores derivados do espaço de cores HSV apresentam as melhores taxas de acerto. Sendo que tal taxa cresce rapidamente até três dimensões e depois cresce mais lentamente. Tal fato mostra que a disper-

são dos vetores favorece tal classificador mesmo quando usadas poucas dimensões.

Também no gráfico da figura 2, nota-se que os classificadores bayesianos tiveram desempenho superior aos classificadores ingênuos de Bayes para a maioria dos casos. Isto mostra, que para este caso, a suposição de independência entre as características não é adequada.

Para os classificadores baseados no perceptron multicamadas, foi necessário o número máximo de iterações. Os EQM obtidos nos treinamentos ficaram acima do mínimo estipulado. Apesar disto, para o caso HSV, foram obtidas taxas de acerto próximas as obtidas com o classificador bayesiano para o mesmo caso. Entretanto, as taxas obtidas com o PMC apresentaram comportamento mais instável como pôde ser observado no gráfico da figura 2. Como foi utilizada apenas uma topologia, não é possível afirmar que o PMC é inadequado para tal tarefa, pois é necessário avaliar outras topologias.

Os resultados mostram que para os espaços de cores adotados, os classificadores apresentam comportamento semelhante, pois para o caso HSV as taxas de acerto crescem rapidamente e para o caso RGB este crescimento é lento. O que mostra que para poucas dimensões os vetores do caso HSV favorecem mais os classificadores que para o caso RGB.

A arquitetura adotada beneficiou-se da capacidade de segmentação não supervisionada do algoritmo JSEG. O conhecimento necessário para identificar os principais elementos das cenas foi embutido no classificador de padrões que foi empregado para classificar os segmentos das imagens e não os *pixels* destas, como é feito normalmente nos trabalhos da área agrícola. Os resultados mostraram que tal arquitetura é viável para a tarefa de análise de cenas em pomares de laranjeiras. Também, com tal arquitetura, é possível considerar novas aplicações, pois esta possibilita que novos elementos sejam reconhecidos, quando o classificador é modificado. Por fim, com o desenvolvimento dos algoritmos de segmentação, tal arquitetura pode ser viabilizada para o uso em aplicações com restrições temporais.

## 10. Conclusões

Neste trabalho, a análise de cenas de pomares de laranjeiras foi feita com uma arquitetura modular, que possibilitou a comparação de diferentes classificadores de padrões e de diferentes tipos de vetores de características. Os classificadores bayesianos treinados com os vetores de características derivados do espaço de cores HSV mostraram-se mais adequados. Tal fato, pos-

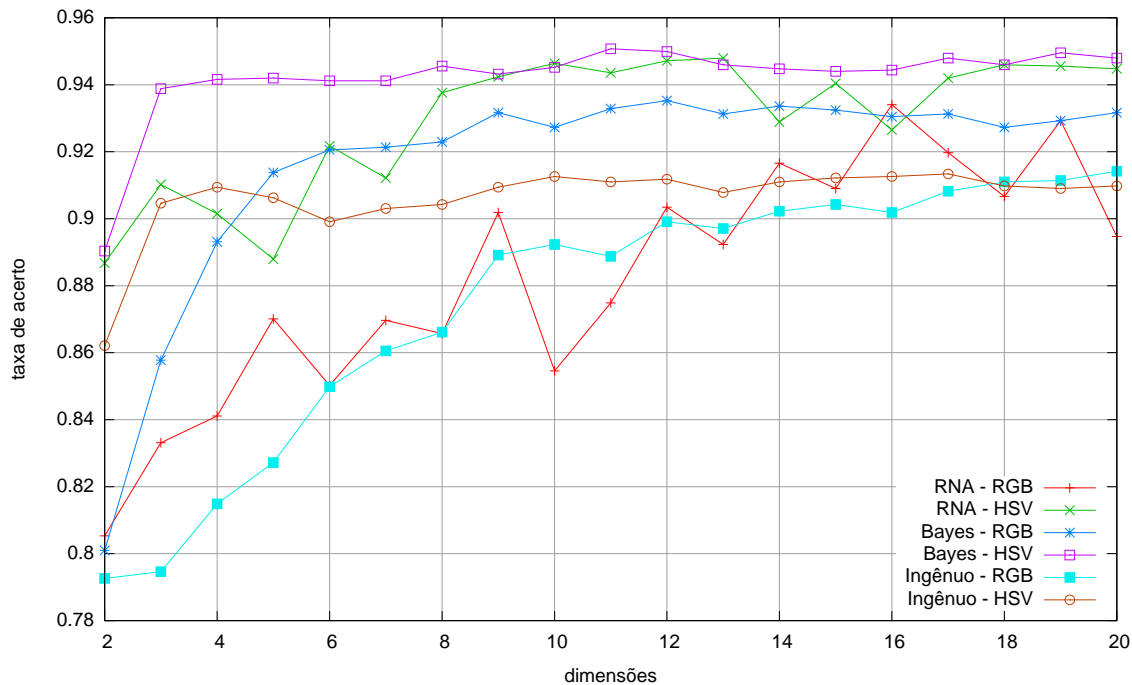


Figura 2. Taxas de classificações corretas para cada quantidade de dimensões.

sibilitou o reconhecimento dos principais elementos dos pomares de laranjeiras com a arquitetura adotada.

## Referências

- [1] P. Annamalai, W. S. Lee, and T. F. Burks. Color vision system for estimating citrus yield in real-time. In *Proceedings... ASAE/CSAE Annual International Meeting*, ASABE, 2004.
- [2] L. da F. Costa and R. M. Cesar Júnior. *Shape analysis and classification - Theory and Practice*. CRC Press LLC, Boca Raton, Florida, EUA, 1 edition, 2001.
- [3] Y. Deng, B. S. Manjunath, and H. Shin. Color image segmentation. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 446–451. IEEE Computer Society, jun. 1999.
- [4] R. O. Duda and P. E. Hart. *Pattern classification and scene analysis*. John Wiley & Sons, Inc., New York, EUA, 1 edition, 1973.
- [5] M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396, 2002.
- [6] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, New York, EUA, 1 edition, 2003.
- [7] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice-Hall, New Jersey, EUA, 2 edition, 1999.
- [8] C. Igel and M. Hüsken. Empirical evaluation of the improved Rprop learning algorithm. *Neurocomputing*, 50(C):105–123, 2003.
- [9] B. S. Manjunath, J. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, 2001.
- [10] J. C. Neto, G. E. Meyer, D. D. Jones, and A. J. Surkan. Adaptive image segmentation using a fuzzy neural network and genetic algorithm for weed detection. In *Proceedings... ASAE Annual International Meeting*, ASABE, jul. 2003.
- [11] P. Paalanen, J.-K. Kamarainen, J. Ilonen, and H. Kälviäinen. Feature representation and discrimination based on gaussian mixture model probability densities – practices and algorithms. *Pattern Recognition*, 39:1346–1358, 2006.
- [12] P. Paalanen, J.-K. Kamarainen, J. Ilonen, and H. Kälviäinen. GMMBAYES matlab toolbox, 2007. Disponível em: <<http://www.it.lut.fi/project/gmmbayes/>>. Acesso em: 11 mar. 2007.
- [13] A. Plebe and G. Grasso. Localization of spherical fruits for robotic harvesting. *Machine Vision and Applications*, 13:70–79, 2001.
- [14] M. Regunathan and W. S. Lee. Citrus fruit identification and size determination using machine vision and ultrasonic sensors. In *Proceedings... ASAE/CSAE Annual International Meeting*, ASABE, 2005.
- [15] B. L. Steward, L. F. Tian, D. Nettleton, and L. Tang. Reduced-dimension clustering for vegetation segmentation. *Transactions of the ASAE*, 47:609–616, 2004.